

Московский государственный технический университет им. Н.Э. Баумана.

Факультет «Информатика и управление»

Кафедра «Системы обработки информации и управления»

Курс «БКИТ»

Отчет по домашнему заданию

Выполнил: студент группы ИУ5-31И Кареникс Артёмс

Москва, 2017 г

Задание:

Разработать программу, реализующую многопоточный поиск в файле.

1. Программа должна быть разработана в виде приложения Windows Forms на языке C#. По желанию вместо Windows Forms возможно использование WPF.
2. В качестве основы используется макет, разработанный в лабораторных работах №4 и №5.
3. Реализуйте функцию поиска с использованием расстояния Левенштейна в многопоточном варианте. Количество потоков для запуска функции поиска вводится на форме в поле ввода (TextBox).
4. Реализуйте функцию записи результатов поиска в файл отчета. Файл отчета создается в формате .txt или .html.

Код программы:**MinMax.cs**

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Text;

namespace LAB_5
{
    class MinMax
    {
        public int Min
        {
            get;
            set;
        }
        public int Max
        {
            get ;
        }
    }
}
```

```

        set ;
    }
    public MinMax(int a, int b)
    {
        Min = a;
        Max = b;
    }

}
}

```

Form1.Designer.cs

```

namespace LAB_5
{
    partial class Form1
    {
        /// <summary>
        /// Обязательная переменная конструктора.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Освободить все используемые ресурсы.
        /// </summary>
        /// <param name="disposing">истинно, если управляемый ресурс должен быть уда-
        лен; иначе ложно.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }
    }
}

```

```

#region Код, автоматически созданный конструктором форм Windows

/// <summary>
/// Требуемый метод для поддержки конструктора – не изменяйте
/// содержимое этого метода с помощью редактора кода.
/// </summary>
private void InitializeComponent()
{
    this.openFileDialog1 = new System.Windows.Forms.OpenFileDialog();
    this.label1 = new System.Windows.Forms.Label();
    this.button1 = new System.Windows.Forms.Button();
    this.button2 = new System.Windows.Forms.Button();
    this.textBox1 = new System.Windows.Forms.TextBox();
    this.textBox2 = new System.Windows.Forms.TextBox();
    this.label2 = new System.Windows.Forms.Label();
    this.label3 = new System.Windows.Forms.Label();
    this.label4 = new System.Windows.Forms.Label();
    this.listBox1 = new System.Windows.Forms.ListBox();
    this.label5 = new System.Windows.Forms.Label();
    this.textBox3 = new System.Windows.Forms.TextBox();
    this.label6 = new System.Windows.Forms.Label();
    this.SuspendLayout();

    //
    // openFileDialog1
    //
    this.openFileDialog1.FileName = "openFileDialog1";
    this.openFileDialog1.FileOk += new
System.ComponentModel.CancelEventHandler(this.openFileDialog1_FileOk);

    //
    // label1
    //
    this.label1.AutoSize = true;
    this.label1.Location = new System.Drawing.Point(17, 20);
    this.label1.Name = "label1";
    this.label1.Size = new System.Drawing.Size(54, 13);

```

```
this.label1.TabIndex = 0;
this.label1.Text = "File Name";
//
// button1
//
this.button1.Location = new System.Drawing.Point(431, 20);
this.button1.Name = "button1";
this.button1.Size = new System.Drawing.Size(75, 25);
this.button1.TabIndex = 1;
this.button1.Text = "Browse";
this.button1.UseVisualStyleBackColor = true;
this.button1.Click += new System.EventHandler(this.button1_Click);
//
// button2
//
this.button2.Location = new System.Drawing.Point(408, 122);
this.button2.Name = "button2";
this.button2.Size = new System.Drawing.Size(92, 30);
this.button2.TabIndex = 2;
this.button2.Text = "Search words";
this.button2.UseVisualStyleBackColor = true;
this.button2.Click += new System.EventHandler(this.button2_Click);
//
// textBox1
//
this.textBox1.Location = new System.Drawing.Point(98, 64);
this.textBox1.Name = "textBox1";
this.textBox1.Size = new System.Drawing.Size(100, 20);
this.textBox1.TabIndex = 3;
//
// textBox2
//
this.textBox2.Location = new System.Drawing.Point(212, 100);
this.textBox2.Name = "textBox2";
this.textBox2.Size = new System.Drawing.Size(22, 20);
```

```
this.textBox2.TabIndex = 4;

//
// label2
//
this.label2.AutoSize = true;
this.label2.Location = new System.Drawing.Point(16, 67);
this.label2.Name = "label2";
this.label2.Size = new System.Drawing.Size(76, 13);
this.label2.TabIndex = 5;
this.label2.Text = "Enter the word";
//
// label3
//
this.label3.AutoSize = true;
this.label3.Location = new System.Drawing.Point(16, 103);
this.label3.Name = "label3";
this.label3.Size = new System.Drawing.Size(190, 13);
this.label3.TabIndex = 6;
this.label3.Text = "Maximal value of Levenshtein distance";
//
// label4
//
this.label4.AutoSize = true;
this.label4.Location = new System.Drawing.Point(17, 164);
this.label4.Name = "label4";
this.label4.Size = new System.Drawing.Size(68, 13);
this.label4.TabIndex = 8;
this.label4.Text = "Found words";
//
// listBox1
//
this.listBox1.FormattingEnabled = true;
this.listBox1.Location = new System.Drawing.Point(98, 164);
this.listBox1.Name = "listBox1";
this.listBox1.Size = new System.Drawing.Size(123, 134);
```

```

this.listBox1.TabIndex = 9;
//
// label5
//
this.label5.AutoSize = true;
this.label5.Location = new System.Drawing.Point(17, 131);
this.label5.Name = "label5";
this.label5.Size = new System.Drawing.Size(120, 13);
this.label5.TabIndex = 10;
this.label5.Text = "Enter number of threads";
this.label5.Click += new System.EventHandler(this.label5_Click);
//
// textBox3
//
this.textBox3.Location = new System.Drawing.Point(153, 124);
this.textBox3.Name = "textBox3";
this.textBox3.Size = new System.Drawing.Size(45, 20);
this.textBox3.TabIndex = 11;
//
// label6
//
this.label6.AutoSize = true;
this.label6.Location = new System.Drawing.Point(268, 131);
this.label6.Name = "label6";
this.label6.Size = new System.Drawing.Size(41, 13);
this.label6.TabIndex = 12;
this.label6.Text = "TIMER";
//
// Form1
//
this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
this.ClientSize = new System.Drawing.Size(518, 307);
this.Controls.Add(this.label6);
this.Controls.Add(this.textBox3);

```

```

        this.Controls.Add(this.label15);
        this.Controls.Add(this.listBox1);
        this.Controls.Add(this.label14);
        this.Controls.Add(this.label13);
        this.Controls.Add(this.label12);
        this.Controls.Add(this.textBox2);
        this.Controls.Add(this.textBox1);
        this.Controls.Add(this.button2);
        this.Controls.Add(this.button1);
        this.Controls.Add(this.label11);
        this.Name = "Form1";
        this.Text = "Form1";
        this.Load += new System.EventHandler(this.Form1_Load);
        this.ResumeLayout(false);
        this.PerformLayout();
    }

```

#endregion

```

private System.Windows.Forms.OpenFileDialog openFileDialog1;
private System.Windows.Forms.Label label11;
private System.Windows.Forms.Button button1;
private System.Windows.Forms.Button button2;
private System.Windows.Forms.TextBox textBox1;
private System.Windows.Forms.TextBox textBox2;
private System.Windows.Forms.Label label12;
private System.Windows.Forms.Label label13;
private System.Windows.Forms.Label label14;
private System.Windows.Forms.ListBox listBox1;
private System.Windows.Forms.Label label15;
private System.Windows.Forms.TextBox textBox3;
private System.Windows.Forms.Label label16;

```

}

}

Form1.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.IO;
using System.Diagnostics;

namespace LAB_5
{
    public partial class Form1 : Form
    {
        List<string> results = new List<string>();

        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
        }

        public List<string> SplitText(string fileName)
        {
            List<string> textByWords = new List<string>();
            File.OpenRead(fileName);
        }
    }
}
```

```

        string text = File.ReadAllText(fileName);
        string[] words = text.Split(' ', '.', ',', '!', '?', '(', ')', '=',
'+', '-', '\n');
        foreach (string temp in words)
        {
            if (!textByWords.Contains(temp))
            {
                textByWords.Add(temp);
            }
        }
        return textByWords;
    }

```

```

private void button1_Click(object sender, EventArgs e)
{
    openFileDialog1.Filter = "Текстовые файлы|*.txt";
    openFileDialog1.ShowDialog();
    label1.Text = openFileDialog1.FileName;
}

```

```

private void openFileDialog1_FileOk(object sender, CancelEventArgs e)
{
}

```

```

public static List<string> searchWords(object obj)
{
    Tuple <List<string>, string, int> obj1 = (Tuple<List<string>, string, int>)
obj;

    string str = obj1.Item2;
    int wordLen=str.Length;
    String word = str.ToUpper();
    int maxDistance = obj1.Item3;
    List<string> tempList = new List<string>();
    foreach (string str1 in obj1.Item1)
    {

```

```

int tempLen = str1.Length;
int distance;

if (wordLen == 0)
{
    distance = tempLen;
}

string temp = str1.ToUpper();

int [,] matrix = new int [wordLen+1 , tempLen+1];
for (int i = 0; i <= wordLen; i++) matrix[i, 0] = i;
for (int j = 0; j <= tempLen; j++) matrix[0, j] = j;

for (int i = 1; i <= wordLen; i++)
{
    for (int j = 1; j <= tempLen; j++)
    {
        int symbEqual = (
            (word.Substring(i - 1, 1) ==
            temp.Substring(j - 1, 1)) ? 0 : 1);

        int ins = matrix[i, j - 1] + 1; //Добавление
        int del = matrix[i - 1, j] + 1; //Удаление
        int subst = matrix[i - 1, j - 1] + symbEqual;

        //Элемент матрицы вычисляется
        //как минимальный из трех случаев
        matrix[i, j] = Math.Min(Math.Min(ins, del), subst);

        if ((i > 1) && (j > 1) &&
            (word.Substring(i - 1, 1) == temp.Substring(j - 2, 1)) &&
            (word.Substring(i - 2, 1) == temp.Substring(j - 1, 1)))
        {

```

```

        matrix[i, j] = Math.Min(matrix[i, j], matrix[i - 2, j - 2]
+ symbEqual);
    }

```

```

    }
}

```

```

    if (matrix[wordLen, tempLen] <= maxDistance)
    {
        tempList.Add(temp + " (" + matrix[wordLen, tempLen] + ")");
    }
}

```

```

    return tempList;
}

```

```

private void button2_Click(object sender, EventArgs e)
{
    List<string> results=new List<string>();
    if (label1.Text == "File Name") MessageBox.Show("Choose file", "Error");
    else if (textBox1.Text.Length == 0) MessageBox.Show("Enter the word",
"Error");
    else if (textBox2.Text.Length == 0) MessageBox.Show("Enter Levenshtein
distance", "Error");
    else if (textBox3.Text.Length ==0) MessageBox.Show("Enter number of
threads", "Error");
    else
    {
        listBox1.Items.Clear();
        results=subArrays(SplitText(label1.Text));
    }
    MessageBoxButtons butons = MessageBoxButtons.YesNo;
}

```

```

        DialogResult YesNo;
        YesNo = MessageBox.Show("Make a report?", "Choose the answer", buttons);
        if (YesNo == DialogResult.Yes)
        {
            makeReport(results);
        }

    }

    private void label15_Click(object sender, EventArgs e)
    {

    }

    public List<string> subArrays(List<string>list)
    {

        int numberOfThreads;
        int.TryParse(textBox3.Text, out numberOfThreads);

        int destination;
        int.TryParse(textBox2.Text, out destination);

        int numberOfelements = list.Count;

        string str = textBox1.Text;

        int numberOfelementsInSubArray = numberOfelements / numberOfThreads;

        List<MinMax> borders = new List<MinMax>();

        Task<List<string>>[] tasks = new Task<List<string>>[numberOfThreads];

        Stopwatch timer = new Stopwatch();

```

```

        for (int i = 0; i < numberOfThreads; i++)
        {
            if ((i+1)!=numberOfThreads)
            {
                MinMax temp = new MinMax(i * numberOfelementsInSubArray, (i + 1) *
numberOfelementsInSubArray - 1);
                borders.Add(temp);
            }
            else
            {
                MinMax temp = new MinMax(i * numberOfelementsInSubArray,
numberOfelements-1);
                borders.Add(temp);
            }
        }

        timer.Start();

        for (int i = 0; i < numberOfThreads; i++)
        {
            List<string> tempList = list.GetRange(borders[i].Min, borders[i].Max-
borders[i].Min);
            tasks[i] = new Task<List<string>>(searchWords, new Tuple<List<string>,
string, int>(tempList,str,destination));
            tasks[i].Start();
        }

        Task.WaitAll(tasks);

        timer.Stop();

        label6.Text = timer.Elapsed.ToString();

        List<string> results = new List<string>();
        for (int i = 0; i < numberOfThreads; i++)
        {

```

```

        listBox1.Items.Add("Поток " + (i+1).ToString() + ":");
        results.Add("Поток " + (i + 1).ToString() + ":");
        foreach (var x in tasks[i].Result)
        {
            listBox1.Items.Add(x.ToString());
            results.Add(x.ToString());
        }
    }

    return results;

}

public void makeReport(List<string> results)
{
    string ReportFileName = "Report_" +
DateTime.Now.ToString("dd_MM_yyyy_hhmmss")+".txt";

    StringBuilder b=new StringBuilder();

    foreach (string x in results) b.AppendLine(x.ToString());

    File.AppendAllText(ReportFileName, b.ToString());

}

}

}

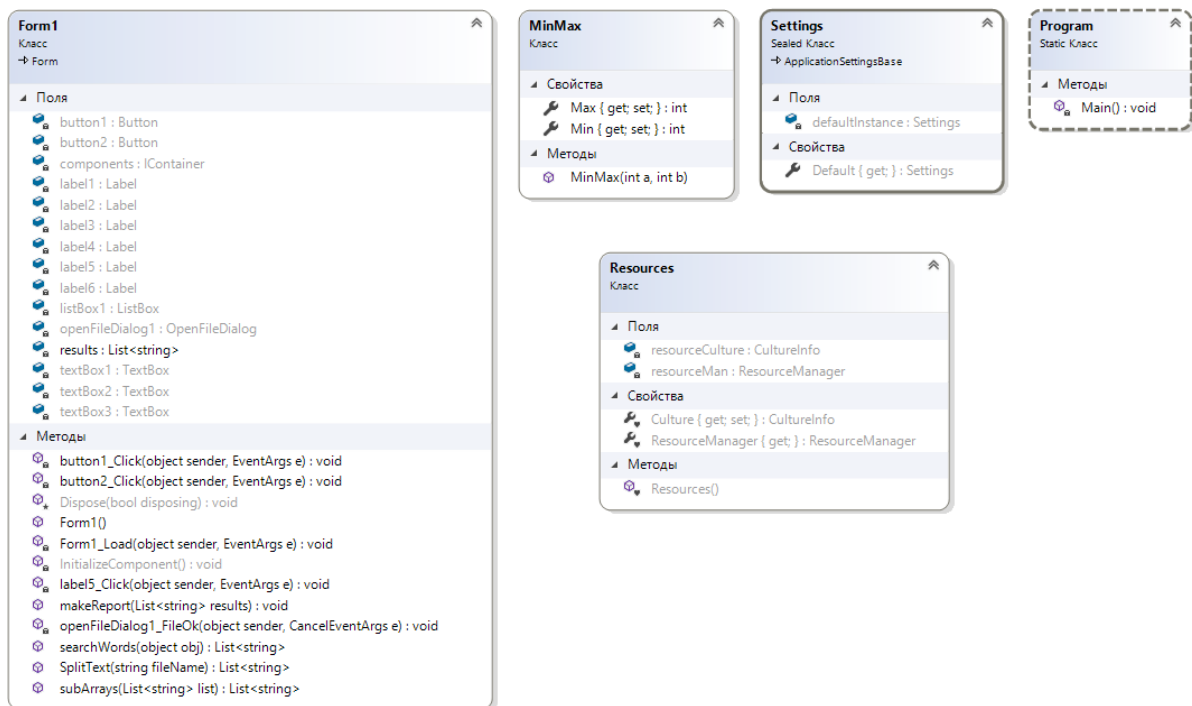
```

Program.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace LAB_5
{
    static class Program
    {
        /// <summary>
        /// Главная точка входа для приложения.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}
```


Диаграмма классов:



Результаты

The screenshot shows the **Form1** application window. At the top, the file path **E:\Univer\BKIT\LAB_4\LAB_4\LAB_4\test.txt** is displayed next to a **Browse** button. Below this, there are input fields for **Enter the word** (containing 'ф'), **Maximal value of Levenshtein distance** (set to 6), and **Enter number of threads** (set to 10). A timer shows **00:00:00.0222247** next to a **Search words** button. The **Found words** section displays a list of results for 'Поток 1:':

- РАБОТ (5)
- И (1)
- ПО (2)
- КУРСУ (5)
- БКИТ (5)
- (1)
- РАБОТА (6)
- №1 (2)
- ДЛЯ (3)

